# SKLEP RTV&AGD

## Jędrzej Ozimek s16760

- **1. Dziedzina problemowa:** System został stworzony z myślą o obsłudze sieci sklepów z urządzeniami elektrycznymi typu Media Markt.
- 2. Cel: System ma na celu ułatwienie relacji między sklepami pracownikami oraz producentami. Ma on też na celu przetrzymywanie danych na temat inwentarzu.
- **3. Zakres odpowiedzialności systemu:** System ma pozwalać na zarządzanie inwentarzem oraz pracownikami.
- 4. Użytkownicy systemu: Pracownicy, Administratorzy.

#### 5. Wymagania użytkownika:

Omówienie struktur

System musi przechowywać dane o produktach. Należy pamiętać o unikatowym ID produktu, nazwie produktu, cenie oraz jego kodzie firmowym wykorzystywanym przez firmę go produkującą do identyfikacji produktu.

Producenci posiadają unikalny identyfikator, nazwę, numer kontaktowy oraz opcjonalny numer konta bankowego. Dzielą się oni na dwie kategorie: Producentów lokalnych oraz zagranicznych. Dla zagranicznych musimy znać ich kraj pochodzenia oraz czy ich produkty są importowane. Dla lokalnych producentów z kolei potrzebny jest ich adres. Producenci są powiązani z produktami za pomocą mapy opartej o kod firmowy danego produktu.

Do celów przechowywania adresu Producentów, Sklepów i pracowników potrzebna jest oddzielna klasa Adres zawierająca informacje o kraju, mieście, ulicy, adresie i ewentualnie numerze mieszkania.

Sklepy muszą pamiętać swoje unikatowe ID, unikatowe dane adresowe oraz rodzaj budynku w którym się znajduje w celach rozliczeń. Dokładniej do wyboru posiadamy zwykły budynek który składa się z nazwy, metrażu, ilości pięter czy jest on wynajmowany oraz jeżeli jest, to za jaką sumę pieniędzy miesięcznie. Drugą opcją jest kompleks który opisuje sklepy znajdujące się w pasażach, galeriach handlowych etc. Kompleks posiada dane dotyczące jego nazwy, miesięcznego czynszu, pietra na którym sklep się znajduje oraz metrażu sklepu.

Sklepy dodatkowo muszą przetrzymywać informacje o stanie inwentarzu poprzez posiadanie odnośników do produktów jak również i ich ilości w danym sklepie. Wiele produktów może być dostępne w wielu sklepach.

Każdy sklep posiada także listę pracowników którzy w nim pracują. Pracownicy mogą pracować tylko w jednym sklepie na raz.

Sklep posiada swój podtyp Sklep z Serwisem który określa sklepy posiadające serwis na miejscu. Dla takich sklepów dodatkowo zapamiętujemy godziny otwarcia serwisu w postaci godziny otwarcia i zamknięcia oraz oddzielną listę pracowników typu serwisant.

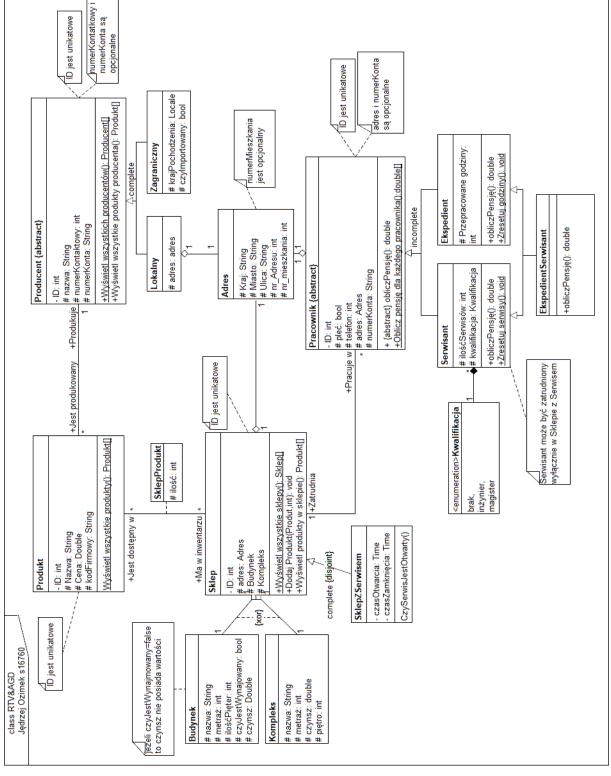
Pracownicy w systemie są opisywani za pomocą unikatowego ID, imienia i nazwiska, płci oraz telefonu. Dodatkowo w celu ułatwienia opłaty pracownik może podać swój adres lub numer konta bankowego. Pracownicy dzielą się na następujące typy: Ekspedient, Serwisant oraz EkspedientSerwisant. W przyszłości mogą pojawić się dodatkowe typy.

Ekspedient musi posiadać informację o przepracowanych godzinach w celu obliczenia pensji. Serwisant z kolei musi posiadać ilość wykonanych serwisów oraz jego kwalifikacje (brak, inżynier, magister). Dodatkowo Serwisanci są ograniczeni do pracowania wyłącznie w Sklepach z Serwisem. EkspedientSerwisant jest sumą poprzednich klas i posiada wszystkie właściwości.

Wszystkie typy pracowników muszą obliczać pensję dla swojej klasy za pomocą ich unikatowych atrybutów.

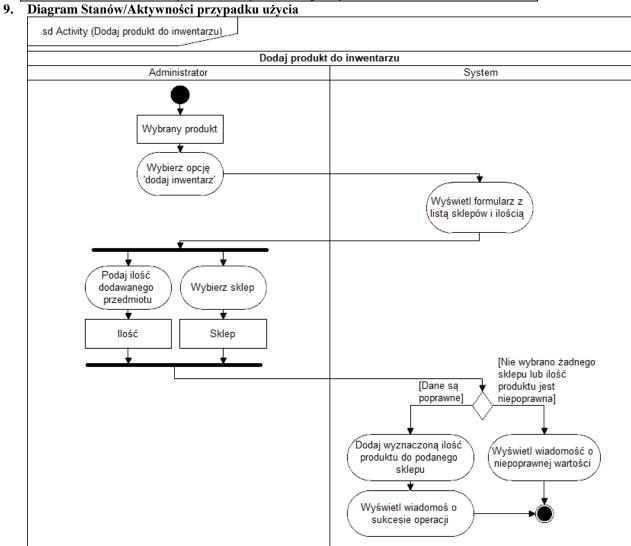
6. Diagram przypadków użycia UC RTV&AGD Jędrzej Ozimek s16760 Wyświetl wszystkie produkty Użytkownik Wyświetl wszystkie produkty od danego producenta <<inlcude>> Wyświetl listę wszystkich producentów <<extend>> Wyświetl wszystkie produkty dostępne w danym sklepie Pracownik <<inlcude>> Wyświetl wszystkie sklepy Oblicz pensję dla pracownika <<include>> Administrator Dodaj produkt do inwentarzu sklepu <<include>> Zlicz pensje dla każdego pracownika Źresetuj ilość serwisów wykonanych przez serwisantów, Koniec miesiąca Zresetuj ilość godzin przepracowanych przez ekspedientów

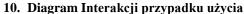
7. Wstępny diagram klas InumerKontatkowy numerKonta są opcjonalne ID jest unikatowe adres i numerKonta ID jest unikatowe są opcjonalne # krajPochodzenia: Locale # czylmportowany: bool +Wyświet! wszystkich producentów(): Producent[] +Wyświet! wszystkie produkt producenta(): Produkt[] numerMieszkania jest opcjonalny Zagraniczny # Przepracowane godziny: +obliczPensję(): double +Zresetuj godziny(): void Acomplete Ekspedient → incomplete # Kraj: String # Miasto: String # Ulica: String # nr\_Adresu: int # nr\_mieszkania: int Producent {abstract} # adres: adres Lokalny Adres - D: int

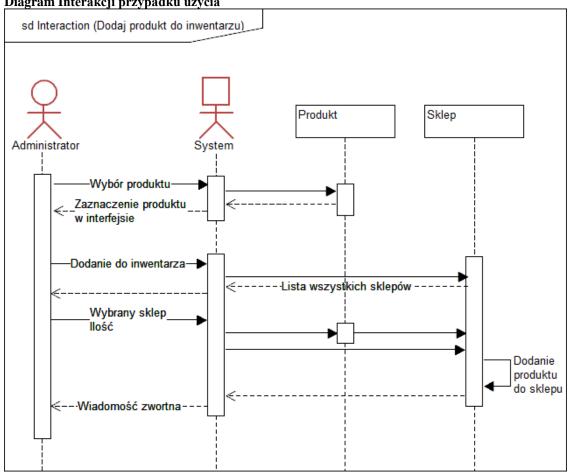


8. \_ Dokumentacja Przypadku użycia

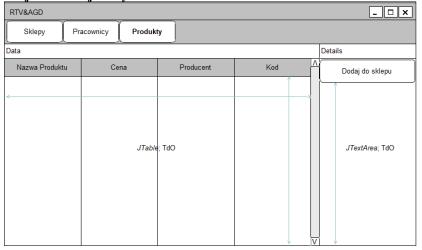
Sekcja	Treść
Nazwa	Dodanie produktu do inwentarzu
Aktorzy	Administrator, System
Scenariusz główny	<ol> <li>Administrator zaznacza produkt który chce dodać i wybiera opcję dodawania inwentarzu</li> <li>System wyświetla listę sklepów i prosi o wybranie jednego z nich oraz o ilość dodawanego inwentarzu.</li> <li>Administrator wybiera sklep z listy i podaje ilość.</li> <li>System dokonuje zmian w inwentarzu sklepu i wyświetla wiadomość o poprawnym wykonaniu operacji</li> </ol>
Scenariusz poboczny 1szy	Administrator nie zaznaczył produktu który chce dodać do sklepu.      1A. System blokuje przycisk od dodawania produktów dopóki administrator nie wybierze produktu
Scenariusz poboczny 2gi	Administrator nie wypełnił pola z ilością inwentarzu.     3A. System wyświetla powiadomienie o braku danych i wychodzi z operacji.







11. Projekt interfejsu użytkownika





# 12. Rozwiązania zastosowanie w projekcie

- Sklep
  - Powiązanie z produktem Produkt posiada tabelę sklepów w którym się znajdują się sklepy w których jest dostępny. Ilość produktów jest przechowywana po stronie sklepu w mapie, której kluczem jest produkt. Dodatkowo posiada on funkcje Add, Increase i Decrease do kontrolowania ilości produktu.
  - **Dodawanie serwisantów** Sklep z serwisem jak i zwykły sklep posiadają funkcję dodaj pracownika, z tą różnicą, że w zwykłym sklepie sprawdzany jest typ pracownika.

#### **Pracownik**

EkspedientSerwisant – Dziedziczy on po Ekspediencie i posiada także interfejs ISerwisant który w pełni mapuje działania Serwisanta i z nim go współdzieli.

### **Producent**

Powiązanie z produktem – Produkt posiada pojedynczy odnośnik do swojego producenta, podczas gdy producent wstawia swoje produkty do mapy gdzie kluczem jest kod Firmowy produktu.

Klasy z unikalnym ID – ID jest deklarowane na podstawie statycznej wartości prywatnej nextID. W momencie użycia konstruktora ID otrzymuje aktualną wartość nextID po czym wartość nextID jest zwiększana o jeden.

13. Końcowy diagram klas

